AN1971
APPLICATION NOTE

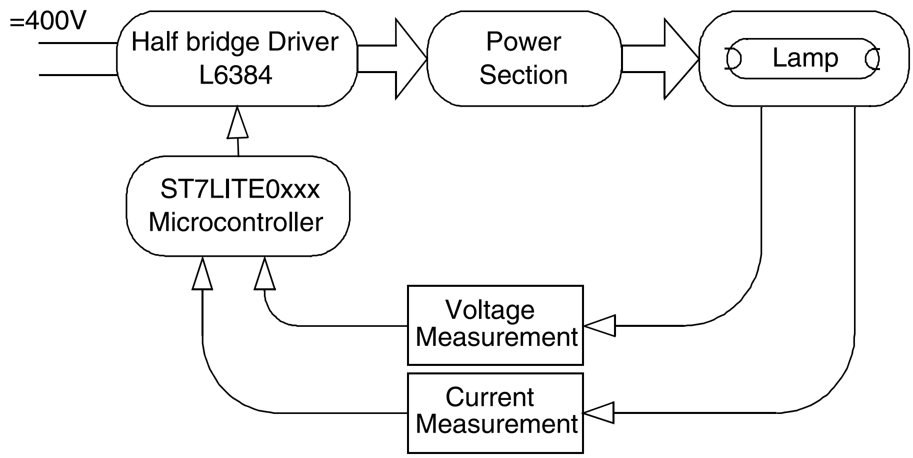
ST7LITE0xxx MICROCONTROLLED BALLAST

Demand for flexibility is increasing in new ballast applications. If a designer can use the same ballast with different tube lamp wattages and types, savings can be made reducing logistic costs. The aim of this application note is to show designers how the ST7 microcontroller helps in the design of such a ballast. In addition, it shows how the use of the ST7LITE0xxx micro-controller adds some attributes facilitating design-work and improving the ballast functionality.

1 INTRODUCTION

Figure 1 shows a diagram of the whole application.

Figure 1. Block Diagram



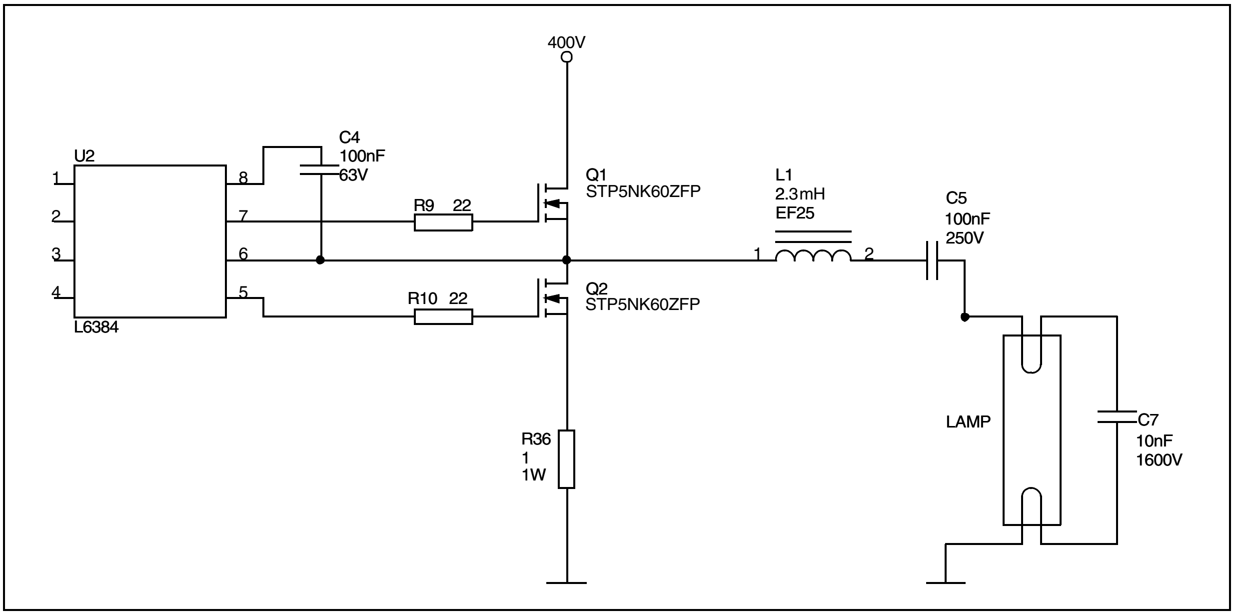
Rev. 3

1.1 POWER SECTION

This application note focuses on the lamp control and therefore a DC voltage 400V has been chosen to supply the application. To have a complete ballast connectable to the standard mains, the EVAL6562-80 board can be chosen to serve as a PFC part, between AC-mains and the 400V DC link.

In this application, voltage-fed series resonant half-bridge inverters are used to drive a fluorescent tube lamp in zero-voltage switching mode and the microcontroller handles the control of the ballast. The microcontroller drives the L6384 high voltage half bridge driver. L6384 is a small eight-pin device, with one input, selectable dead time and implemented bootstrap diode (refer to the L6384 datasheet for more information). In Figure 2, the driver and resonant tank topology are shown. Thanks to the microcontroller flexibility, existing resonant circuits can also be used.

Figure 2. Power Section Circuit

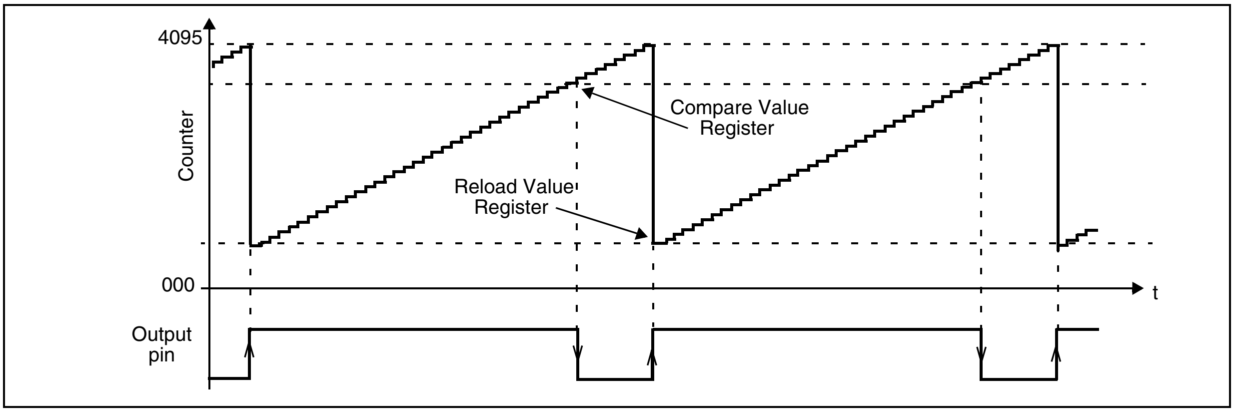


1.2 DIGITAL SECTION

The ST7LITE0xxx microcontroller has been chosen as it is small and easy to use (please refer to Section 5 REFERENCES AND RELATED MATERIALS [1]). This processor has a 1.5kB flash program memory, 128 bytes of RAM and moreover it has 128 bytes of EEPROM of usable memory, for example, for storing lamp parameters. There is no need for any external clock circuitry, because it has an integrated RC oscillator with an accuracy of 1%. This oscillator provides a clock signal up to 8MHz. What makes this microcontroller a strong tool for the ballast application are the peripherals: Autoreload Timer, Lite Timer and Analog to Digital Converter. Foremost, it must be highlighted that the software drivers for all the peripherals are available in the ST7 software library (see ST7 Software Library, http://www.st.com/mcu) and you do not need to spend time to develop your own.

The Autoreload timer is a peripheral which controls a PWM output from the microcontroller. The principle of its function is shown in Figure 3. At the heart of the autoreload timer is a free-running counter, which works absolutely independently from the processor core. For designers, there are only two important values: “Reload Value Register” and “Compare Value Register”. The counter increments its value to the maximum. When it is reached, it switches the output pin to the logical 1 (5V) and after that the counter starts incrementing again from the value stored in the already mentioned “Reload Value Register”. After each increment, the counter is compared to the “Compare Value Register”. If the match occurs, the output pin is switched to the logic value 0 (0V).

Figure 3. PWM Function



From Figure 3, it can be seen that the control of the PWM signal is very simple with ST7LITE0xxx through just the two registers: by writing into the Reload Register, you select the frequency and by writing into the Compare Register you can select the duty cycle. This way, the frequency can be selected from 2kHz up to 4MHz. The incremental period change is 125ns with fclock = 8 MHz.

In addition to the autoreload timer, there is also another generic timer available. This timer (called Lite Timer) is a free running counter generating a software interrupt every 1ms. There is a simple software trick using a variable which counts the number of these interrupts. If you want an event to occur after a certain time (e.g. switch from preheating to ignition mode after one second), you should watch this global variable and when it reaches the desired value a proper procedure is run. The time from 1ms up to the dozen of minutes can be measured this way (with the step of 1ms).

To connect the analog world to the digital core there is an analog to digital converter (ADC) implemented in the ST7LITE0xxx. This ADC has two input ranges, the first measures the analog voltage from 0 to 5V in order to obtain a digital value ranging from 0 to 255 (8-bit resolution). The second turns on an integrated amplifier with a gain of eight, which means that it can measure the voltage in the range from 0 to 250mV. This integrated amplifier is very useful, especially when measuring the small voltage drop on the current sense resistors.

Lamp current and voltage must be measured to have complete information about the ballast circuitry. In Figure 4 and Figure 5 are the circuits used to filter the voltage on the current sense resistor. The first filter is used to obtain the peak current value and the second one to get an average current.

Figure 4. Peak Current Sensor

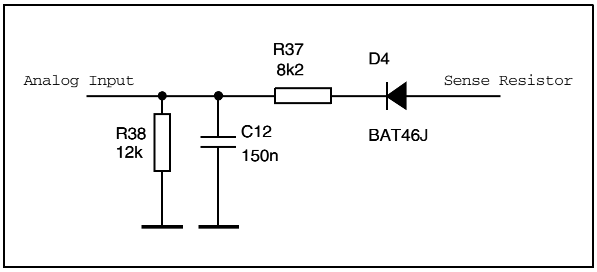
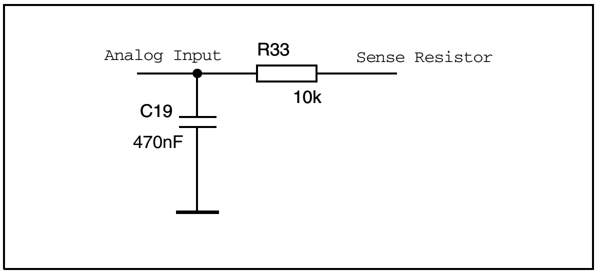


Figure 5. DC Current Sensor



Two circuits are used for voltage measurement, similar to the current measurement. The first, used to measure a peak value, is depicted in Figure 6. It is a simple voltage divider with output in the range from 0 to 5V. To avoid an error caused by the voltage drop on the diode D3, the divider has been split into two parts. The second measurement circuit, used to obtain the voltage DC offset on the lamp, is shown in Figure 7. Because the offset can be either positive or negative the circuit has been adapted to raise the zero point to 2.5V. This means that a measured voltage smaller than 2.5V results in a negative offset and voltage higher than 2.5V gives a positive offset.

Figure 6. Peak Voltage Sensor

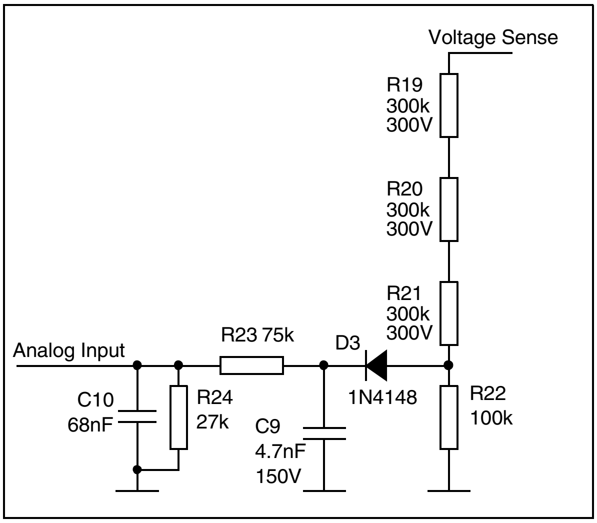
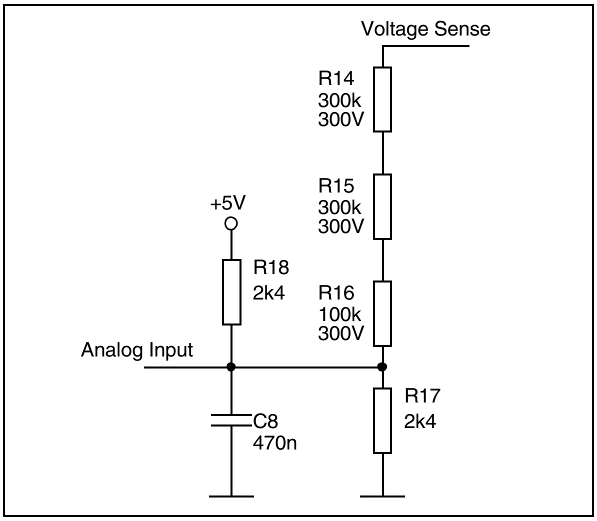


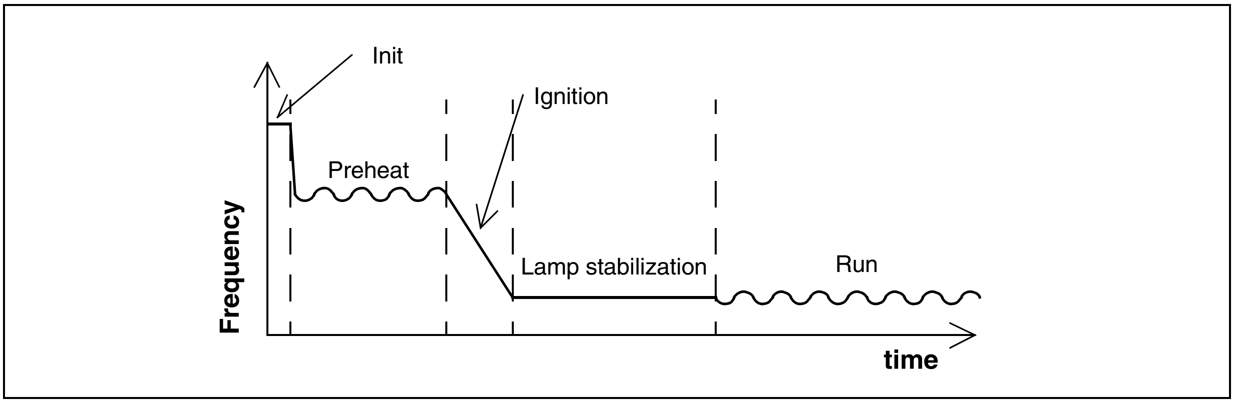
Figure 7. DC Voltage Sensor



2 BALLAST STATES

The ballast goes through different states, from the system power-on to steady state running, End of Life or system power down. An explanatory timing diagram (not in scale) of all the states is shown in Figure 8. Some of the states are common to the classical ballast applications; some are improved thanks to the microcontroller solution.

Figure 8. Frequency Timing Diagram



2.1 INIT STATE

During the init phase, the microcontroller sets the PWM frequency at 100kHz and keeps it steady for 200ms. This feature has been implemented to charge up the blocking capacitor (C5). In Figure 9 and Figure 10 there are the situations with and without this feature. When the application goes directly into the preheat phase (as is usual in systems without the microcontroller), the voltage on the lamp can exceed maximum values due to a premature ignition. Of course, if you want to skip this phase, it is still possible just by changing a single line of code.

Figure 9. Lamp Voltage with Init State

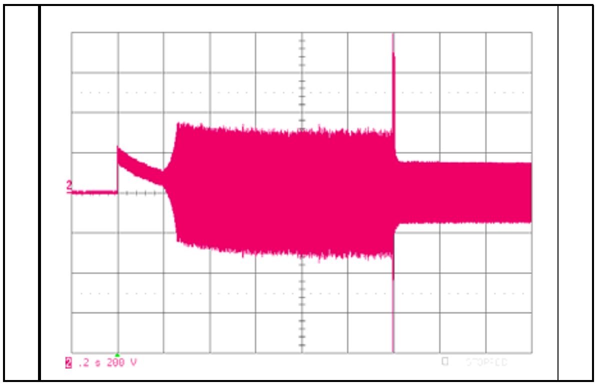
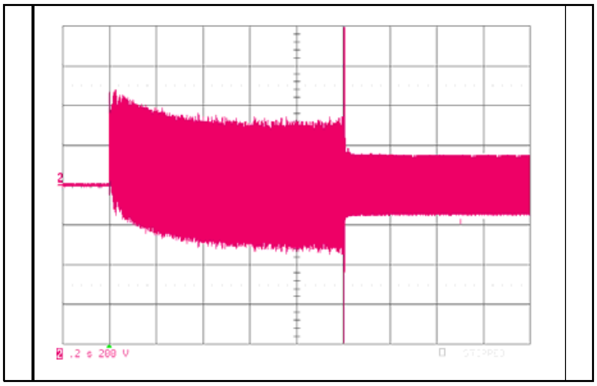


Figure 10. Lamp Voltage without Init State



2.2 PREHEAT STATE

Software has full control during the whole procedure in the preheat state. This means that you select the desired preheating current and then a control algorithm regulates the current. The algorithm used is the simplest one: the step-regulation. The software raises the frequency by one step (125ns) every time the measured current is bigger than the desired value, and vice versa.

In addition, it is possible, by changing only one constant, to set the preheating time from zero (cold start) up to a few seconds.

2.3 IGNITION STATE

The purpose of this state is obvious, the software decrease the frequency (increases the voltage) to ignite the lamp.

As an extra feature, you can adjust the ignition speed (the time between each frequency step) in the software. To detect the moment when the lamp ignites, a simple assumption is used that the voltage across the lamp will significantly decrease after ignition. After detecting the ignition the software moves to the next phase.

2.4 LAMP STABILIZATION STATE

After the ignition of the lamp, a constant frequency is set. You should preselect this frequency in the software as an expected frequency value for the lamp used. This state serves for stabilization of all the lamp characteristics as well as the sensor circuits.

2.5 RUN STATE

At this final phase, the software measures the current to get a given power in the lamp. The calculation of the active power flowing through the lamp is very simple, because the DC current flowing through the sense resistor is the current from the supply. The supply is presumed to be a constant 400V and so the power is linearly proportional to the DC current. Then the measured value is compared with a preset value and consequently the software tries to correct it. Again, the same step regulation control algorithm as in the preheat phase is used.

There is no need to have high speed of this control. Since the gas inside the tube has a long response time, it needs a few milliseconds to stabilize after each change; the speed can be relatively slow (less than a few hertz).

2.6 STOP STATE

If any abnormal conditions occur (see Section 3) the software will automatically switch itself into this safety state. The main function is to turn off the half bridge driver by grounding L6384 SD/DT pin. Because the lamp is then off, there is no reason to consume needless microcontroller energy. That is why ST7LITE0xxx uses the so-called HALT mode. In this mode, the processor core and oscillator are turned off to minimize the consumption (consumption in HALT mode is a few µA).

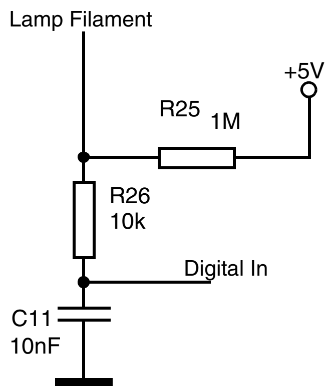
There are two ways to “wake up” the processor from HALT mode. The first one is to turn off and on the mains power supply. The second method is using the external interrupt feature on pin PA7. This feature, briefly described, wakes up the microcontroller when a falling edge (from 5V to 0V) appears on the pin. In Figure 11 the circuit is shown, which detects the lamp insertion, connected to the PA7 pin. After waking up, the microcontroller resets itself and the process starts again from initialization state.

3 ABNORMAL CONDITION HANDLING

3.1 LAMP REMOVAL

This is the most usual error condition during the ballast life. It is relatively common that a lamp is changed without turning off the mains switch. So the detection of a lamp removal and consequent insertion must be implemented. In Figure 11 a detection circuit is shown. When a lamp is present in the application the voltage on the processor input pin is zero (logical 0), while if the lamp is removed the voltage on the input pin rise up to 5V (logical 1). A simple bit-check is enough to detect the lamp presence and then switch the micro to Stop mode whenever it detects the lamp removal.

Figure 11. Detection Circuit for Lamp Removal/Insertion



3.2 LAMP FAILS TO START

This condition can occur only during the ignition state, when the voltage rises without any sign of lamp ignition. That can be caused for example by filament damage or an old lamp. There are three security protections implemented. None of them, however, adds a single component into the application because all of them use the existing possibilities and opportunities given by ST7LITE0xxx. During the ignition phase, the software checks whether the current or the voltage exceeds the preset value. If so, it will immediately stop the driver and switch into the stop state. In addition to these, two protections rather than one were implemented to prevent the sense circuit failure. This protection allows only a limited number of frequency steps in the ignition state. In the event that both current and voltage sensors fail and this protection is not implemented, the frequency will decrease along the resonant curve and the ballast could be damaged.

3.3 RECTIFYING EFFECT (END OF LIFE)

When the lamp is getting old, it highlights an imbalance between filament depletion. It causes a non-zero average voltage across the lamp, which can be measured by the voltage sense circuits depicted in Figure 7. You must determine the maximum allowable level of this offset in the software. In other words, the circuit in Figure 7 is using the microcontroller as a window comparator.

As is usual for all abnormal conditions the software will switch to the Stop state to wait for the lamp change, if this contravention occurs.

3.4 CURRENT OR VOLTAGE EXCEED MAXIMUM VALUES

Exceeding current or voltage maximum (pre-selected in software) in the ballast application can always mean many different problems, but none of them are good for the ballast itself. So in the case that the software detects exceeding of maximum values it rather switches to the Stop state to prevent any damage to the ballast.

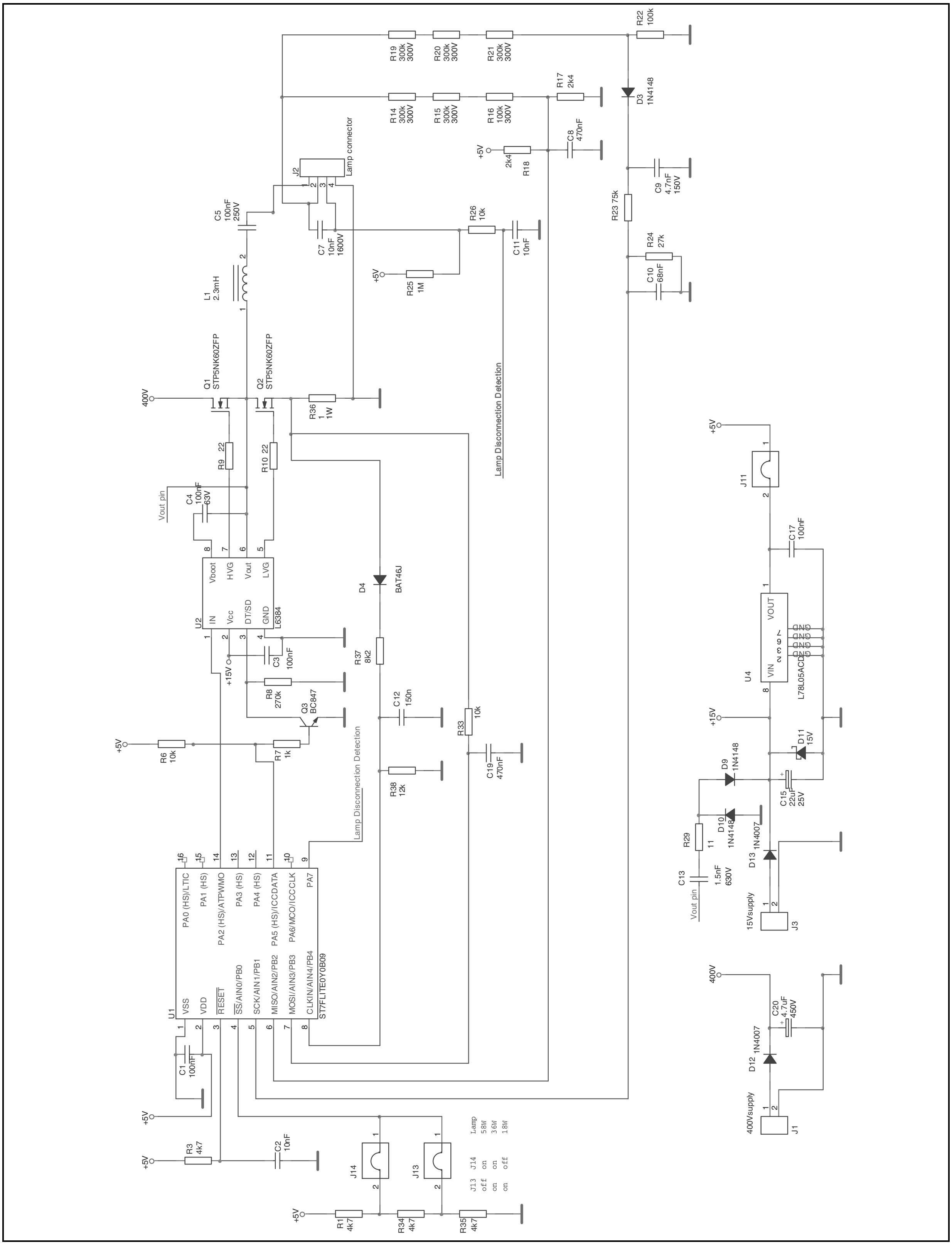
4 APPLICATION BOARD

On the application board there are two jumpers (J13 and J14) used to select the proper microcontroller parameters for the lamp power or lamp type selection. Care must be taken that the ballast is programmed for the nominal lamp power, because if a wrong lamp is selected, the microcontroller will always try to provide a selected power (for example 58W into an 18W tube lamp) which could damage the lamp.

4.1 OUTLOOK

The microcontroller solution with closed loop control allows in principle the detection of the lamp type or power of the lamp inserted. Since this feature requires very detailed know-how of lamp physics it is not implemented in the present version of this demo. An experienced ballast designer however will be able to implement this feature and avoid the limitations described above.

4.2 SCHEMATICS



4.3 COMPONENTS

Item	Quantity	Reference	Part
1	1	C15	22uF
2	1	C20	4.7uF
3	2	C8,C19	470nF
4	5	C1,C3,C4,C5,C17	100nF
5	1	C12	150nF
6	1	C10	68nF
7	3	C2,C7,C11	10nF
8	1	C9	4.7nF
9	1	C13	1.5nF
10	3	D3,D9,D10	1N4148
11	1	D4	BAT46J
12	1	D11	15V
13	2	D12,D13	1N4007
14	1	J1	400Vsupply
15	1	J2	Lamp connector
16	1	J3	15Vsupply
17	3	J11,J13,J14	JUMPER1
18	1	L1	2.3mH
19	2	Q1,Q2	STP5NK60ZFP
20	1	Q3	BC847
21	1	R25	1M Ω
22	5	R14,R15,R19,R20,R21	300k Ω
23	1	R8	270k Ω
24	2	R16,R22	100k Ω
25	1	R23	75k Ω
26	1	R24	27k Ω
27	1	R38	12k Ω
28	3	R6,R26,R33	10k Ω
29	1	R37	8k2 Ω
30	4	R1,R3,R34,R35	4k7 Ω
31	2	R17,R18	2k4 Ω
32	1	R7	1k Ω
33	2	R9,R10	22 Ω
34	1	R29	11 Ω
35	1	R36	1 Ω
36	1	U1	ST7FLITE0Y0B09
37	1	U2	L6384
38	1	U4	L78L05ACD

5 REFERENCES AND RELATED MATERIALS

[1] ST7LITE0xY0, ST7LITESxY0 datasheet

[2] L6384 datasheet

[3] AN1501 Simple Microcontrolled Ballast

[4] ST7 Software Library, downloadable from http://www.st.com/mcu

6 REVISION HISTORY

Date	Revision	Changes
13-Sep-2004	1	Initial release
18-Jan-2006	2	Figure 2 modified Values of components modified in Figure 6 and Figure 7 In Section 2.1, reference to C6 as a blocking capacitor removed Schematics in Section 4.2 updated Component list in Section 4.3 updated
17-Dec-2012	3	Modified part numbers and title to include ST7LITE0xxx.